International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 3 Issue 5, August 2019 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Bank's Roled Based Member Access Control

War War Myint, Hlaing Phyu Phyu Mon, Pa Pa Win, Zin Mar Naing

Faculty of Information Science, University of Computer Studies, Meiktila, Myanmar

How to cite this paper: War War Myint | Hlaing Phyu Phyu Mon | Pa Pa Win | Zin Mar Naing "Bank's Roled Based Member Access Control" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-3 | Issue-5, August 2019, pp.1151-1155, **https://doi.org/10.31142/ijtsrd26533**

IJTSRD26533

ABSTRACT

Today, Banks are essential things for finance. Because services served by banks: transferring money from one place to another, saving money many ways and others, bank's functions are very important. All data concerned with bank are kept to be secure because of the financial cases. And bank's staffs' roles and permissions according to their positions are also important. If staffs' duties and responsibilities are identified properly and correctly, daily round of bank services can be operated efficiently. So, duties and responsibilities of bank staff to be truly served, duties should be assigned by their roles and permissions. Managing the staff, assigning duties and keeping bank's confidential records effectively is a big hurdle these days. In this case, RBAC (role-based access control) is the best way for controlling security of staffs' duties. RBAC is the standard innovation access control model and most important access control model and provides a great way to full fill the access control needs. In this study, bank staffs' duties are controlled to be able to secure with RBAC. There are several factors that are related to the system, and the main ones are: users, organization, positions, roles, tasks, processes, and rules or permissions. The design architecture is based on RBAC concepts, according to the concept, only the administrator has the privilege to manage or administer the data. She/he provides all types of privileges required to maintain users, their authorization and access, and the authorized resources. The administrator controls the largest information, including access to the bank's staffs' files and has the sole access to all potential staffs and their assigned duties. This study took into the account the security access control, and security policies and methods integrated into the RBAC which is appropriate for managing system of Bank. The goal of this system intends to apply Role-based Access Control on bank transaction process.

KEYWORDS: *RBAC concepts; Security Policy; Banking Service Management*

1. INTRODUCTION

The principal motivations behind RBAC are the ability to articulate and enforce enterprise-specific security policies and to streamline the typically burdensome process of security management. RBAC represents a major advancement in flexibility and detail of control from the present-day standards of discretionary and mandatory access control. The policies enforced in a particular stand-alone or distributed system are the net result of the precise configuration of the various components of RBAC. RBAC framework provides administrators with the capability to regulate who can perform what actions, when, from where, in what order, and in some cases under what relational circumstances.

Security is a major concern in today's digital world. Role-based access control provides a mechanism for protecting the digital information in an organization by assigning roles to the individual user and giving permissions to the assigned roles for accessing any resources. This paper describes the importance of roles in an organization and the evolutionary changes that occur with respect to the organizational roles. Here the role is defined as an entity and the attributes of the roles have been identified with their related operations.

RBAC is the standard and most important access control model and has been the most important research topic for the last two decades. Role-Based Access Control model provides a great way to fulfill the access control needs. An access control policy is a statement which specifies the rules about who can access the resources and how much access is given to each user. In RBAC main Focus is on Role. The main idea behind the RBAC is that a role is an intermediate module between users and permissions. In RBAC roles are assigned to the users (many-to-many assignments) and permissions are associated with each role (many-to-many assignments), and thus indirectly assigns users to permissions. [6]

For efficiency, roles can be structured hierarchically so that some roles inherit permissions from others. RBAC simplifies access control compared with the administrative burden that would be required for a direct mapping from individual users to access control lists attached to resources. Once roles with their permissions have been defined, user provisioning simply requires that office staff assign users to roles as authorized by management.

In RBAC system data access is provided to the user according to their role. The roles are mapped to access permissions and users are mapped to appropriate roles. The Administrator assigned the roles to users based on their responsibilities and qualifications in their organization. Permissions are assigned to roles as per their qualifications instead of users. In RBAC, a role hierarchy structure is used.

The roles can inherit permissions from other roles. The RBAC system provides flexible control and management by having two mappings of the user to role and roles to privileges on data objects.

The majority of these components also constitute the building blocks of policies of other RBAC implementations. Note that many RBACs support policies that may contain constructs, such as role hierarchies. While these can be expressed in RBAC in an alternative way, appropriate extensions may include in this policy components, so maintaining their applicability to other RBAC models.

Knowledge of policy structure is vital for programmers, policy administrators and users. It is important for programmers because the structure presented here can serve as a template or a reference implementation. Policy structure is also reflected by the serialized forms of policies. Policy administrators need a good knowledge of policy structure since it helps them to understand the methods that are used to modify policies.

This is especially important when such policy components are used to restrict access to policies themselves. Users need a limited knowledge of the policy components to protect themselves and their information. Many RBAC implementations allow users to interact with the access control enforcer, thus permitting the users to select the roles they want to enter, to select the rules that should be used for role entry or authorization, or to select the credentials or prerequisites that should be considered in an access control decision.

If the user has an idea of "what is going on" user can assign a trust level to the policy enforcing application, and based on such a trust level the user might hide or provide credentials that are needed to enter a specific role.

2. PRIVILEGES AND ROLES

A possible extension to the above privileges supports for privilege inheritance. This hierarchy can be easily simulated in RBAC, but having explicit support for privilege hierarchies allows for more compact policy specifications. If a parent and a child privilege are in a hierarchical relation, then the parent privilege can be used as a prerequisite in all the method calls where the child privilege is accepted as a prerequisite.

By using hierarchical relation between privileges we can express privilege dependencies, for example, a write privilege can only be assigned to a role with a read privilege [12].

It must be possible to convert the parent privilege with its parameters into the child privilege, which can have a different set of parameters. This mapping looks almost like a policy rule that treats a senior privilege as a prerequisite and authorizes inherited privileges automatically (i.e. a user has no control over this change).

By introducing rules that allow privileges as prerequisites the difference between roles and privileges starts to disappear. Indeed, privileges could even be considered as roles that cannot be prerequisites; the separation of the notion of privileges and roles is merely a convenience for administrators.

There are models, for example, PERMIS [15], that do not differentiate privileges from roles; indeed, there are access control models – like the capability-based access control – that handle roles and privileges as if they were one concept. Hierarchical privileges are basically functional roles with a restriction that should not be used as prerequisites to organizational roles. A useful extension to RBACs could be proved that use organizational roles only; however, all RBACs could benefit from a clean separation of functional and organizational roles, even in the form of privilege dependencies [13, 14]. Role and privileges of this banking system can be described as follow.

Table 2.1 Role Table

Role ID	Role Name	Description
R_1	Admin	Administrator
R_2	Board of Director	Owner or administrator
R_3	Head Office Manager	Manager
R_4	Branch Manager	Manager
R_5	Remittance Counter Checker	Counter Checker
R_6	Withdraw Counter Checker	Counter Checker
R_7	Deposit Counter Checker	Counter Checker
R_8	Remittance Operator	Operator
R_9	Withdraw Operator	Operator
R_10	Deposit Operator	Operator

Table 2.2 Types of Permission Table

Permission ID	Permission Name	Description
P_1	Read	View individual transaction
P_2	Write	Fill data deals with account transactions
P_3	Edit	Edit deposit amount
P_4	Delete	Cancel wrong transactions
P_5	Read report	Detail and summary report(monthly, annual) by branch
P_6	Execute	Calculate interest amount
P_7	Approve	Check transactions step by step
P_8	Read all	Read all transactions
P_9	Execute	Define User Account, Trace Login/ logout time

3. Table 2.3 Permission Transactions Table

Role ID	Permission ID	Department
R_1	P_9	Administration
R_2	P_5	Administration
R_3	P_1, P_5, P_7, P_8	Administration
R_4	P_1, P_5, P_7, P_8	Administration
R_5	P_1, P_7,	Remittance
R_6	P_1, P_7,	Withdraw
R_7	P_1, P_7,	Deposit
R_8	P_1, P_2, P_3, P_4	Remittance
R_9	P_1, P_2, P_3,P_6	Withdraw
R_10	P_1, P_2, P_3,P_6	Deposit

4. AUTHORIZATION RULES

Authorization rules are among the most critical components of a policy from the point of view of both access control and self-administration. A role with privileges to modify rules can control the assignment of privileges to policy roles, which may even include the role itself, thus it is possible that a role has the power to modify its own privileges [14].

The basic idea underlying the bank's whole process management technology is a system that explains the roles and manages the activities of workflow through the use of designing software. In this study, an administrator manages the system in such a way that it can control the employee using the RBAC. The RBAC is used to define organization membership of the individual working in the bank by assigning individuals to roles, assign permissions to roles, and now activate the job function or role with respect to the appropriate points in the sequence. Each user is assigned one or more "ROLES" and each "ROLE" is assigned one or more "PERMISSION" that is authorized for the users in that role by the administrator. Permission consists principally of the opportunity to perform operations within an activity of the bank. Objects, such as files and processes, can be organized into hierarchies. In such object hierarchies, it is important to know not only the access of role group to an object but also to know whether the path in the hierarchy could be traversed [11].

4. MODEL DATABASE FOR THE BANKING MANAGEMENT

In the bank, all the bank information and process information is personal and are stored in the database system. The data must only be accessed by users who are defined or authorized. This is the first step for any information stored and any secured data. To make the queries simple and provide an easy to administer, a decent database must reduce data redundancy.

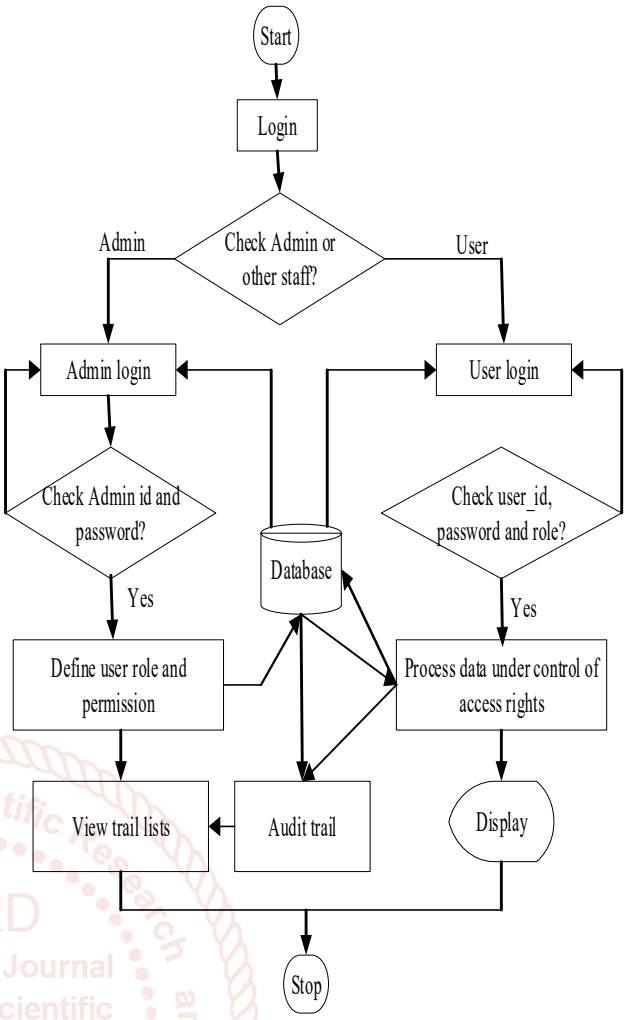

Figure 4.1 System Flow Diagram

The banking system is implemented with RBAC in this study. In this system, different types of tables that are related are stored in a database which allows data to be accessed means to control restricting access according to authorization rule and activation rule. The system has mainly two sides; administration side and user side.

In the user side, user side can be differentiated with roles. These roles are defined into hierarchical role design according to the application domain banking system. Each role has each privilege to be granted to access their assets in each part. Each responsibility of each role depends on whether their permission to be allowed or not to access objects or data sets. One role might own one or more user in this system. It depends on how system administrator to decide how to be set according to job requirements. The following figure is shown the role levels and permissions of staffs (users) and admin.

4.1 ROLE LEVEL OF SYSTEM

The diagram shows the role level hierarchies of the banking system. There are five role levels which are Board of Director, Head Office Manager, Branch Manager, Counter Checker and Operator by banking organization structure. These five role level have each access rights to access allowed transactions such as remittance transaction, deposit transaction and withdraw transaction.

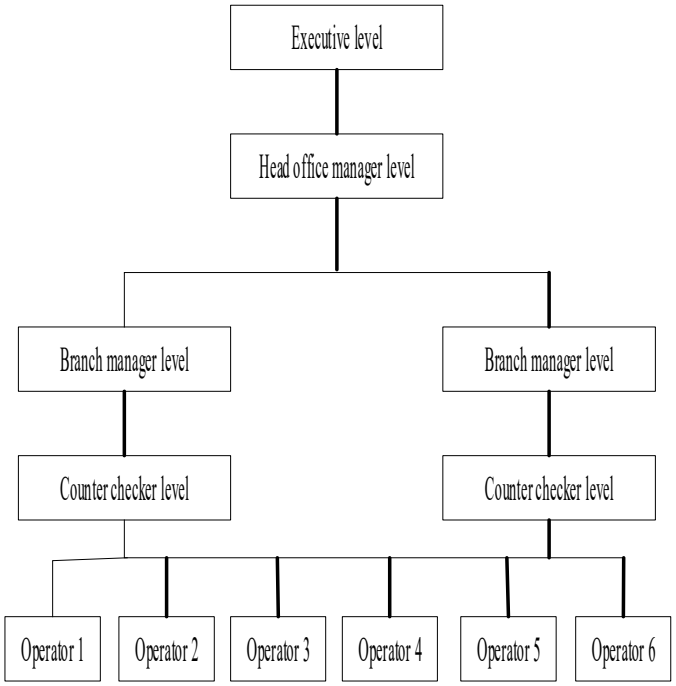

Figure 4.2 Role Hierarchy of the System

5. CONCLUSION

Information system in a bank has unique specific security and privacy requirements. If management would like to apply RBAC in this information system to reduce the administrative tasks and manage the smooth running of the organization, then must be adopted. Other issues that should be looked at are control of data sharing in an open distributed environment. We, therefore, propose this model of RBAC; this approach increases data availability, confidentiality, integrity, accountability, which is the most important requirement in the company management. In addition, we have also proposed a program that takes permission evaluation when conflicting roles are present.

Managing the bank has become a long way from serving principally as a means of making it easier to manage access to applications. As the growing number of employees‟ rules-driven projects indicates, RBACs are increasingly likely to address critical management objectives such as greater cost efficiencies, improved compliance, and reduce security exposure. Working as part of an integrated, automated role-management and identity-management solution, RBAC can go a long way toward helping avert potential management catastrophes in increasing collaborative and complex company environments.

6. ACKNOWLEDGMENTS

I would like to take this opportunity to express my sincere thanks to all my senior associates who gave me a lot of valuable advice and information. I am also grateful to all respectable people who directly or indirectly contributed to the success of this research. I especially also thank my parents. I owe my respectful thanks to Dr. Mie Mie Khin, Rector of the University of Computer Studies, Meiktila for allowing me to develop this research and giving me general guidance during the period of studying time.

I owe a great debt of gratitude to Dr. Hlaing Phyu Phyu Mon, Professor and Head of Faculty of Information Science, University of Computer Studies, Meiktila, for giving advice.

REFERENCE

[1] Ms. Sunita, Prachi, “Efficient Cloud Mining Using RBAC (Role-Based Access Control) Concept”, Volume 3, Issue 7, July 2013.

[2] Edwin Okoampa Boadu, Gabriel Kofi Armah, “Role-Based Access Control (RBAC) Based In Hospital Management”, Volume 3, Issue 9 (September 2014), PP.53-67.

[3] Suganthy. A Dr. T. Chithralekha (Associate Prof.), “Role-Evolution in Role-based Access Control System”, July 2017, ISSN: 2278-9359 (Volume-6, Issue-7).

[4] LIU Dongdong, XU Shiliang, ZHANG Yan, TAN Fuxiao, NIU Lei, ZHAO Jia, “Role-based Access Control in Educational Administration System”, MATEC Web of Conferences 139, 00120 (2017).

[5] Nicola Zannone, “Role Based Access Control (RBAC)”.

[6] Anthony Rhodes, William Caelli, “A Review Paper Role-Based Access Control”.

[7] Ed Coyne, Timothy R. Weil, “ABAC and RBAC: Scalable, Flexible, and Auditable Access Management”, 1520-9202/13/$31.00 © 2013 IEEE.

[8] H.B. Klasky, P. T. Williams, S. K. Tadinada, B. R. Bass ORNL, “A Role-Based Access Control (RBAC) Schema for REAP”, September 2013.

[9] “A best practice case implementing Role-Based Access Control at ABN AMRO, KCP first European Identity Management Conference Munich, May 7-10.

[10] T. Finin, A. Joshi, L. Kagal, Niu, R. Sandhu, W. Winsborough, “ROWLBAC - Representing Role-Based Access Control in OWL”, SACMAT’08, June 11–13, 2008, Estes Park, Colorado, USA.

[11] Hui Qi, Hongxin Mat, Jinqing Li and Xiaoqiang Di ” Access Control Model Based on Role and Attribute and Its Applications on Space-Ground Integration Networks” IEEE 2015.

[12] Cecilia Ionita and Sylvia Osborn. Privilege administration for the role graph model. In Research Directions in Data and Applications Security, IFIP WG 11.3 Sixteenth International Conference on Data and Applications Security, July 28-31, 2002, Kings College, Cambridge, U.K., volume 256 of IFIP International Federation for Information Processing, pages 15–25. Kluwer Academic Publishers, 2003.

[13] Liang Chen “Analyzing and Developing Role-Based Access Control Models”, 2011.

[14] Role-based access control policy administration, March 2004 URL: *http://www.cl.cam.ac.uk/TechReports/*UCAM-CL-TR-586.pdf

[15] David W. Chadwick and Alexander Otenko. The PERMIS X.509 role-based privilege management infrastructure. In Seventh ACM Symposium on Access Control Models and Technologies (SACMAT’02), pages 135–140. ACM Press, 2002.

[16] Michael D. Schroeder Jerome H. Saltzer. The protection of information in computer systems. IEEE, 63(9):1278–1308, September 1975.

[17] Michael Hitchens and Vijay Varadharajan. Tower: A language for role-based access control. In Policies for Distributed Systems and Networks, International Workshop (POLICY'01), Bristol, UK, pages 88–107, 2001.

[18] Ravi Sandhu and Pierrangela Samarati. Access control: Pinciples and practice. IEEE Communications Magazine, 32(9):40–48, 1994.

[19] Ravi Sandhu. Roles versus groups. In Proceedings of the First ACM Workshop on Role-Based Access Control (RBAC'95), pages I–25–26, 1995.

[20] Ravi Sandhu, Edward Coyne, Hal L. Feinstein, and Charles E. Youman. Role-based access control models. IEEE Computer, 29(2):38–47, 1996.

[21] Ravi Sandhu. Role activation hierarchies. In Proceedings of the Third ACM Workshop on Role-Based Access Control (RBAC'98), pages 33–40, 1998.

[22] Ravi Sandhu and Qamar Munawar. How to do discretionary access control using roles. In Proceedings of the Third ACM Workshop on Role-Based Access Control (RBAC'98), pages 47–54, 1998.

[23] Ravi Sandhu, David Ferraiolo, and Richard Kuhn. The NIST model for role-based access control: towards a unified standard. In Proceedings of the Fifth ACM Workshop on Role-Based Access Control (RBAC'00), pages 47–63, 20.